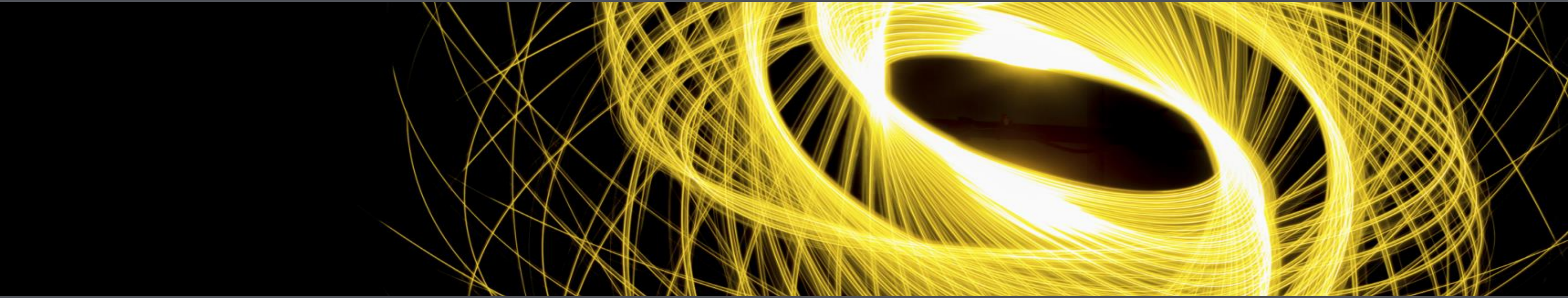
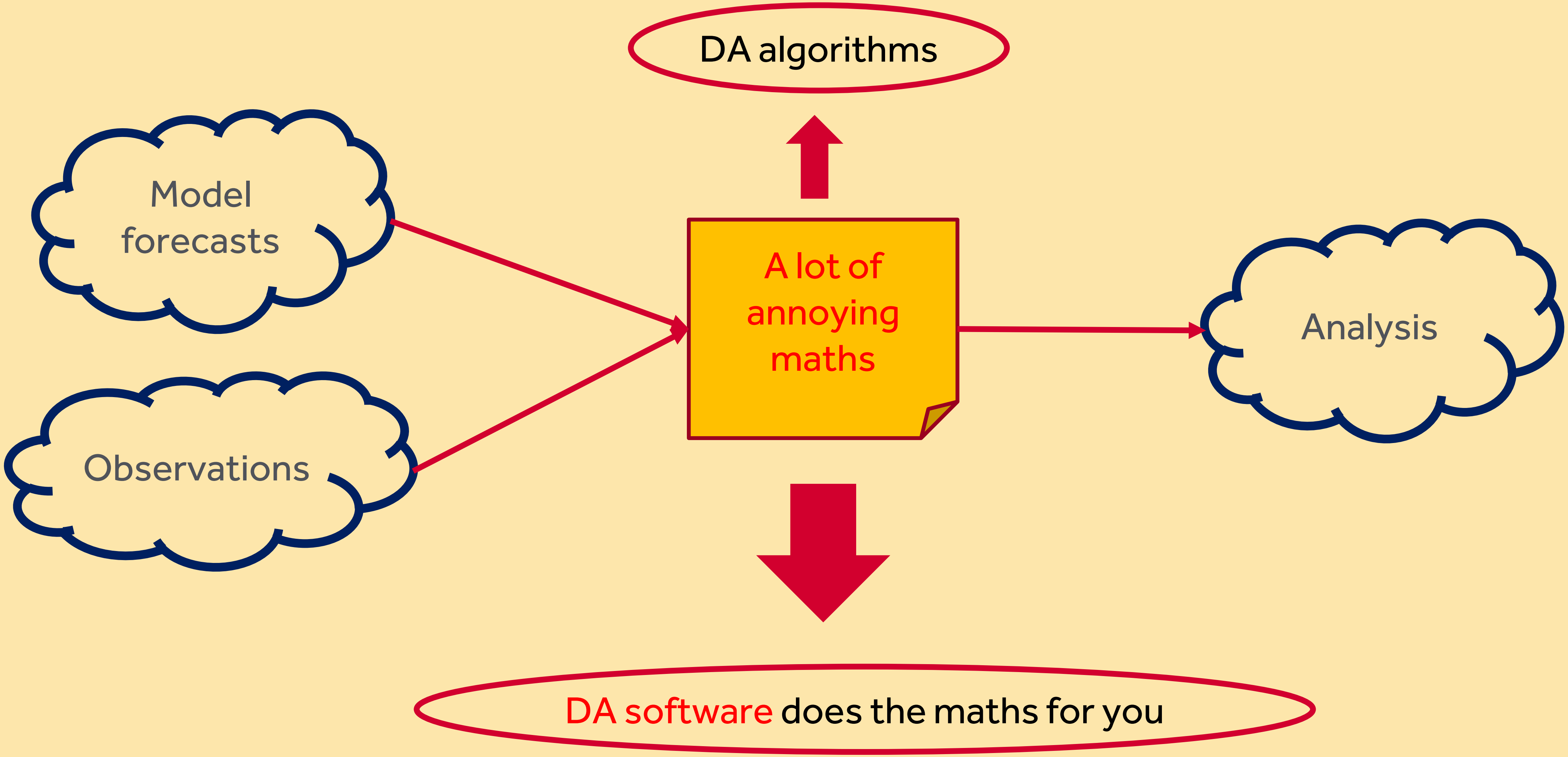
Department of Meteorology
Data assimilation research centre (DARC)

Data assimilation software



Yumeng Chen
yumeng.chen@reading.ac.uk


DA algorithms
Model forecasts
Observations
A lot of annoying maths
Analysis
DA software does the maths for you

University of Reading

When do we want to write our own code?

- Not very complicated algorithms
- Learning details of an algorithm
- Sense of control
- Easier modifications
- Tailored to our own applications
- Fewer dependent libraries

...

$$\mathcal{J}(\mathbf{x}_0) = \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}^b)^{\mathrm{T}}\mathbf{B}^{-1}(\mathbf{x}_0 - \mathbf{x}^b) + \frac{1}{2}\sum_{i=0}^{N}(\mathcal{H}_i(\mathbf{x}_i) - \mathbf{y}_i)^{\mathrm{T}}\mathbf{R}_i^{-1}(\mathcal{H}_i(\mathbf{x}_i) - \mathbf{y}_i)$$

Algorithm 1.2 4D-Var in its basic form

$j = 0$, $\mathbf{x} = \mathbf{x}_0$
while $||\nabla J|| > \epsilon$ **or** $j \leq j_{\max}$
  (1) compute J with the direct model M and H
  (2) compute ∇J with adjoint model $\mathbf{M}^{\mathrm{T}}$ and $\mathbf{H}^{\mathrm{T}}$ (reverse mode)
  gradient descent and update of $\mathbf{x}_{j+1}$
  $j = j + 1$
end

Source: Data Assimilation: Methods, Algorithms, and Applications by Maëlle Nodet, Marc Bocquet, Mark Asch

Why bother with DA software?

- Convenience
  - Prefer np.mean to writing loops
- Optimised and efficient
- Well-maintained and reliable
- Focus on the scientific questions
- Ensures reproducible and consistent scientific research

$$\mathcal{J}(\mathbf{x}_0) = \frac{1}{2}(\mathbf{x}_0-\mathbf{x}^b)^{\mathrm{T}}\mathbf{B}^{-1}(\mathbf{x}_0-\mathbf{x}^b) + \frac{1}{2}\sum_{i=0}^{N}(\mathcal{H}_i(\mathbf{x}_i)-\mathbf{y}_i)^{\mathrm{T}}\mathbf{R}_i^{-1}(\mathcal{H}_i(\mathbf{x}_i)-\mathbf{y}_i)$$

Algorithm 1.2 4D-Var in its basic form

$j=0$, $\mathbf{x}=\mathbf{x}_0$
while $\|\nabla J\| > \epsilon$ **or** $j \le j_{\max}$
  (1) compute J with the direct model M and H
  (2) compute ∇J with adjoint model $\mathbf{M}^{\mathrm{T}}$ and $\mathbf{H}^{\mathrm{T}}$ (reverse mode)
  gradient descent and update of $\mathbf{x}_{j+1}$
  $j = j+1$
end

Source: Data Assimilation: Methods, Algorithms, and Applications by Maëlle Nodet, Marc Bocquet, Mark Asch

Why bother with DA software?

One may even argue that we can build a DA system without a team using good software.

A non-exhaustive list of DA software/libraries/framework

Name	Developers	Purpose (approximately)
DART	NCAR	General
PDAF	AWI	General
JEDI	JCSDA (NOAA, NASA, ++)	General
OpenDA	TU Delft	General
EMPIRE	Reading (Met)	General
ERT	Statoil	History matching (Petroleum DA)
PIPT	CIPR	History matching (Petroleum DA)
MIKE	DHI	Oceanographic
OAK	Liège	Oceanographic
Siroco	OMP	Oceanographic
Verdandi	INRIA	Biophysical DA
PyOSSE	Edinburgh, Reading	Earth-observation DA

Name	Developers	Notes
DAPPER	Raanes, Chen, Grudzien	Python
SANGOMA	Conglomerate*	Fortran, Matlab
hIPPYlib	Villa, Petra, Ghattas	Python, adjoint-based PDE methods
FilterPy	R. Labbe	Python. Engineering oriented.
DASoftware	Yue Li, Stanford	Matlab. Large inverse probs.
Pomp	U of Michigan	R
EnKF-Matlab	Sakov	Matlab
EnKF-C	Sakov	C. Light-weight, off-line DA
pyda	Hickman	Python
PyDA	Shady-Ahmed	Python
DasPy	Xujun Han	Python
DataAssim.jl	Alexander-Barth	Julia
DataAssimilationBenchmarks.jl	Grudzien	Julia, Python
EnsembleKalmanProcesses.jl	Clim. Modl. Alliance	Julia, EKI (optim)
Datum	Raanes	Matlab
IEnKS code	Bocquet	Python

Courtesy:
Patrick Raanes on https://github.com/nansencenter/DAPPER/blob/master/README.md

A non-exhaustive list of DA software/libraries/framework

Name	Developers	Purpose (approximately)
DART	NCAR	General
PDAF	AWI	General
JEDI	JCSDA (NOAA, NASA, ++)	General
OpenDA	TU Delft	General
EMPIRE	Reading (Met)	General
ERT	Statoil	History matching (Petroleum DA)
PIPT	CIPR	History matching (Petroleum DA)
MIKE	DHI	Oceanographic
OAK	Liège	Oceanographic
Siroco	OMP	Oceanographic
Verdandi	INRIA	Biophysical DA
PyOSSE	Edinburgh, Reading	Earth-observation DA

Operational use/research for large models

- UK Met Office is transitioning towards JEDI
- PDAF has been used to investigate marine ecosystem DA, Arctic sea ice DA, etc.
- DART has been used with WRF,MPAS-Atmosphere, CAM

Courtesy:
Patrick Raanes on https://github.com/nansencenter/DAPPER/blob/master/README.md

A non-exhaustive list of DA software/libraries/framework

Methodology research for small models like Lorenz 96

- We used DAPPER to test a simplified DA algorithm (a smoother algorithm)
- IEnKS code was used for idealised parameter estimation tests
- We can still learn a lot with small models

Name	Developers	Notes
DAPPER	Raanes, Chen, Grudzien	Python
SANGOMA	Conglomerate*	Fortran, Matlab
hIPPYlib	Villa, Petra, Ghattas	Python, adjoint-based PDE methods
FilterPy	R. Labbe	Python. Engineering oriented.
DASoftware	Yue Li, Stanford	Matlab. Large inverse probs.
Pomp	U of Michigan	R
EnKF-Matlab	Sakov	Matlab
EnKF-C	Sakov	C. Light-weight, off-line DA
pyda	Hickman	Python
PyDA	Shady-Ahmed	Python
DasPy	Xujun Han	Python
DataAssim.jl	Alexander-Barth	Julia
DataAssimilationBenchmarks.jl	Grudzien	Julia, Python
EnsembleKalmanProcesses.jl	Clim. Modl. Alliance	Julia, EKI (optim)
Datum	Raanes	Matlab
IEnKS code	Bocquet	Python

Courtesy:
Patrick Raanes on https://github.com/nansencenter/DAPPER/blob/master/README.md

DAPPER

University of Reading

Data Assimilation with Python: a Package for Experimental Research (DAPPER)

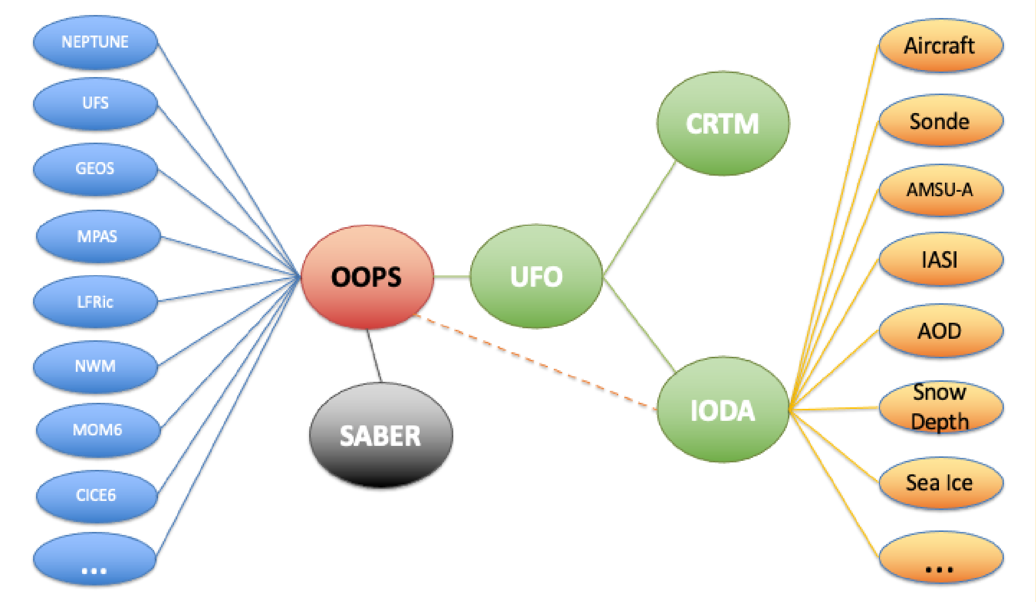
- DAPPER is a set of templates for DA methods
- The typical set-up is a synthetic (twin) experiment as what we do in practicals
- Ease of adding new DA methods and models
- Pure Python implementation
- This can be used to compare different DA algorithms and applications with low-dimensional state vectors

Method	Literature reproduced
EnKF [1]	Sakov08, Hoteit15, Grudzien2020
EnKF-N	Bocquet12, Bocquet15
EnKS, EnRTS	Raanes2016
iEnKS / iEnKF / EnRML / ES-MDA [2]	Sakov12, Bocquet12, Bocquet14
LETKF, local & serial EAKF	Bocquet11
Sqrt. model noise methods	Raanes2014
Particle filter (bootstrap) [3]	Bocquet10
Optimal/implicit Particle filter [3]	Bocquet10
NETF	Tödter15, Wiljes16
Rank histogram filter (RHF)	Anderson10
4D-Var	
3D-Var	
Extended KF	
Optimal interpolation	
Climatology	

https://github.com/nansencenter/DAPPER

Joint Effort for Data assimilation Integration (JEDI)

- Various operational centres opt for it including UKMO, NOAA, etc
- Mainly designed for numerical weather forecast
- Well-maintained software with a large community
- Modern software development workflow
- Designed with flexibility in mind
  - Many components of DA system can be configured in YAML files without the need changing JEDI code



Joint Effort for Data assimilation Integration (JEDI)

- OOPS: Object Oriented Prediction System:
  - 3D-Var; 4DEnsVar; 4DVar; Weak constraint 4DVar
  - LETKF, LGETKF
- UFO: Unified Forward Operator
  - Obs. Operator
  - Quality control and variational bias correction
- SABER: System Agnostic Background Error Representation
  - computing and working with the background error covariance matrix
- IODA: Interface for Observation Data Access
  - handle an immense amount of data from the providers

OOPS

$$\mathcal{J}(\mathbf{x}_0) = \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}^b)^{\mathrm{T}}\mathbf{B}^{-1}(\mathbf{x}_0 - \mathbf{x}^b) + \frac{1}{2}\sum_{i=0}^{N}(\mathcal{H}_i(\mathbf{x}_i) - \mathbf{y}_i)^{\mathrm{T}}\mathbf{R}_i^{-1}(\mathcal{H}_i(\mathbf{x}_i) - \mathbf{y}_i)$$

SABER UFO IODA

PDAF Parallel Data Assimilation Framework

University of Reading

Parallel Data Assimilation Framework (PDAF)

- Focuses on ensemble DA
- Suitable for weather and climate models, e.g. AWI-CM, MITgcm, MPI-ESM, NEMO, etc.
- Implementation is in general efficient and reliable
- Good protocols and well-documented to be used with any models and observations

EnKF → (ETKF, ESTKF, EnKF, SEIK, SEEK)

Nonlinear filtering → (NETKF, Particle filter)

3DVar → (3DVar, 3DEnVar, Hyb3DVar)

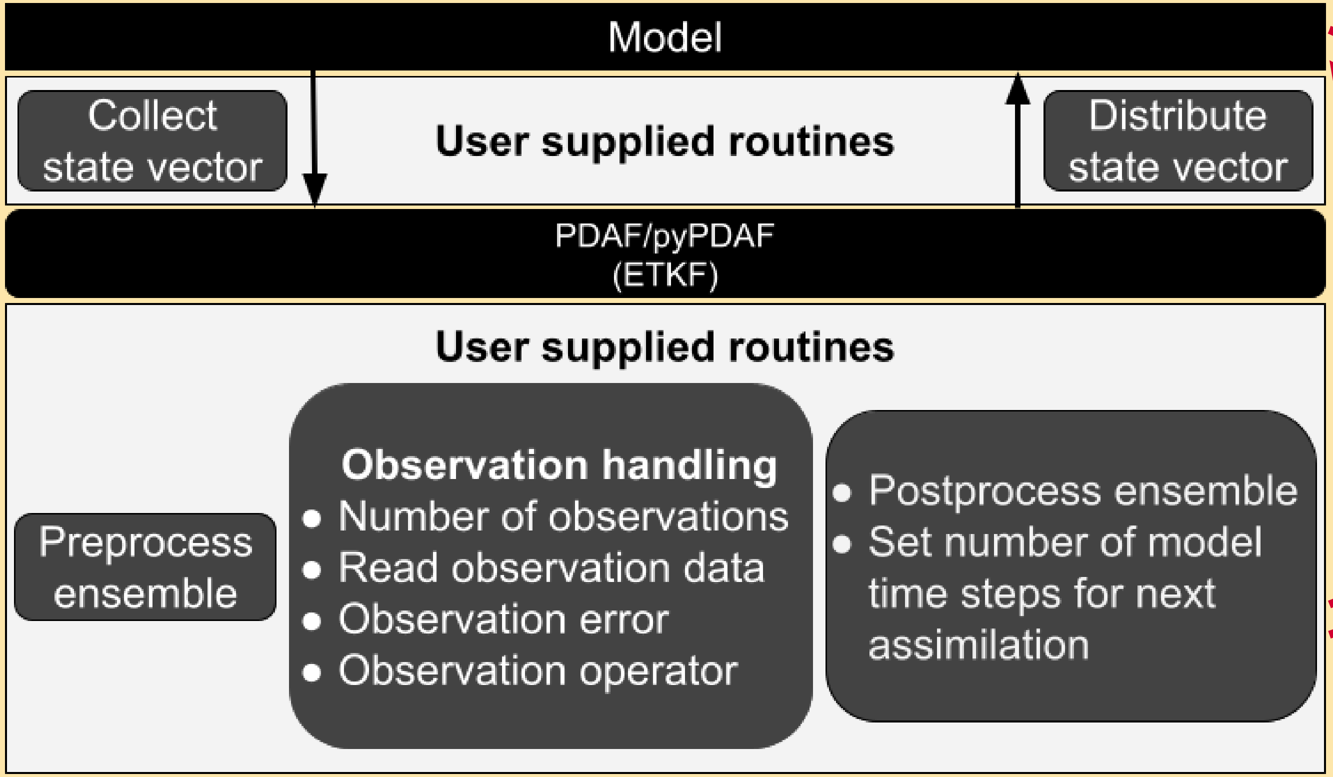
Global filter	Local filter	Smoother
ETKF	✓	✓
ESTKF	✓	✓
EnKF	✓	✓
SEIK	✓	✓
SEEK		
NETKF	✓	✓
Particle filter		
3DVar		
3DEnVar		
Hyb3DVar		

PDAF Parallel Data Assimilation Framework

University of Reading

Parallel Data Assimilation Framework (PDAF)

- Flexibility of PDAF relies on user-supplied routines (grey boxes)



$$\mathbf{x}_{\mathrm{a}}^{i} = \mathbf{x}_{\mathrm{b}}^{i} + \mathbf{K}^{i}\left(\mathbf{y}^{i} - h(\mathbf{x}_{\mathrm{b}}^{i})\right)$$

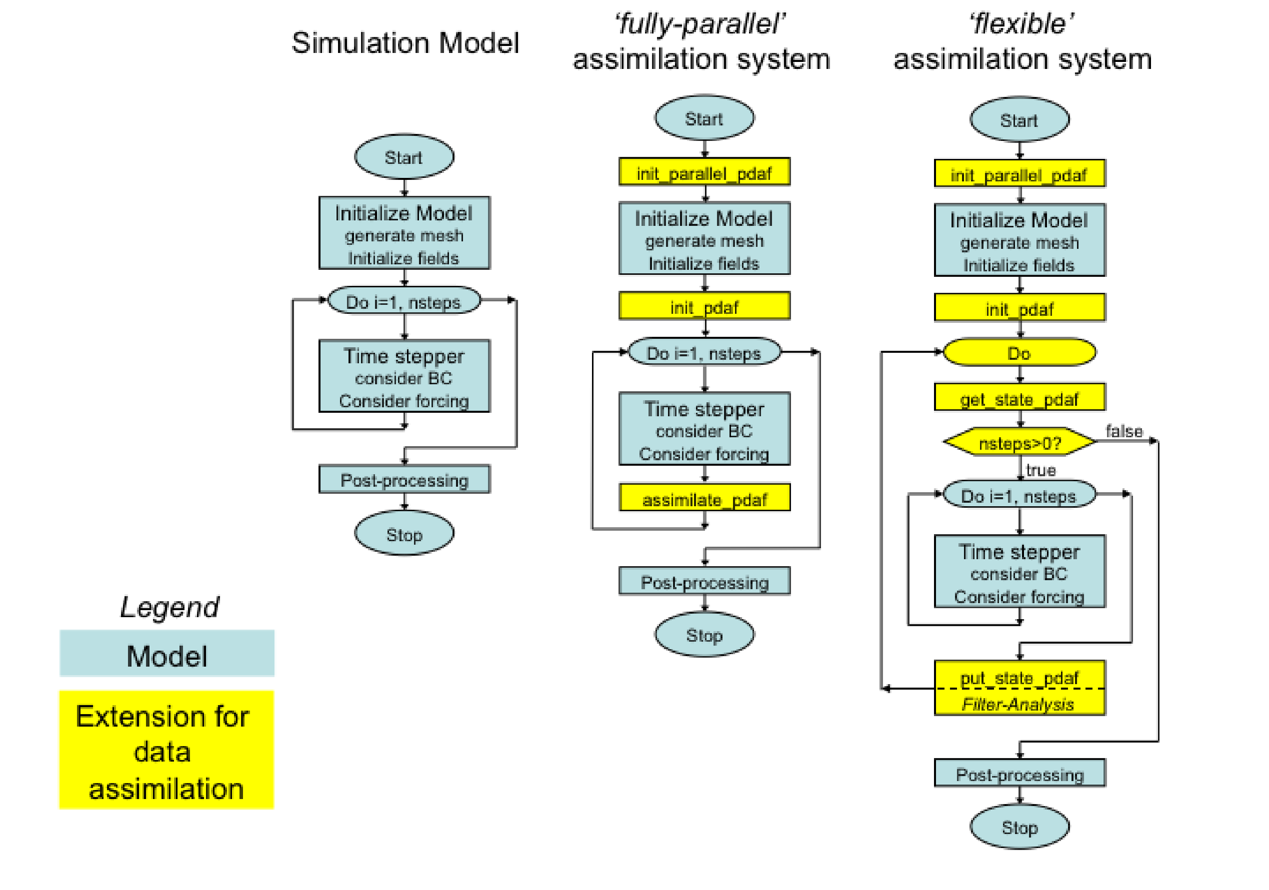
- Types of user-supplied routines vary for different filters; check the PDAF website for references.

PDAF Parallel Data Assimilation Framework

University of Reading

Parallel Data Assimilation Framework (PDAF)

- Coupling to an existing model



PDAF Parallel Data Assimilation Framework

University of Reading

Parallel Data Assimilation Framework (PDAF)

- We now also have a Python interface to PDAF, pyPDAF:
  - https://github.com/yumengch/pypdaf
- Build a simple DA system using pyPDAF:
  - https://colab.research.google.com/github/yumengch/pyPDAF/
  - If you run the notebook on your local computer:
    - conda create -n pyPDAF -c yumengch -c conda-forge pyPDAF jupyter matplotlib
  - Any feedback on pyPDAF and the tutorials are welcome!